Intel® System Studio 2015 Update 1 Installation Guide and Release Notes

Installation Guide and Release Notes for Windows* Host

Document number: 329488-008US

16 December 2014

Contents

1 Introduction 4
Change History 5
Intel® System Studio 2015 Update 1 5
Intel® System Studio 2015 5
Intel® System Studio 2014 Update 2 6
Intel® System Studio 2014 Update 1 6
Intel® System Studio 2014 6
Intel® Software Manager 7
Product Contents 8
2 Getting Started 9
3 Technical Support and Documentation 9
Release Notes Location 9
Article & Whitepaper Locations 11
Support 11
4 What's New 12
Update 1 12
Initial Release 12
5 System Requirements 14
Supported Host Platforms 14
Eclipse* Integration Prerequisites 14
Host Prerequisites and Resource Requirements 15

5.1.1 Host Space Requirements by Component 15
5.1.2 Intel® Integrated Performance Primitives (Intel® IPP) Details 15
5.1.3 Intel® C++ Compiler 15
Target Software Requirements 16
Target Prerequisites and Resource Requirements 16
5.1.4 Target Space Requirement by Component 16
5.1.5 Intel® VTune™ Amplifier target OS kernel configuration 17
5.1.6 Intel® VTune™ Amplifier Feature vs. Resource Matrix 18
Hardware Requirements 18
6 Installation Notes 19
Installing the Tool Suite 19
Using the online installer 20
6.1.1 Online Installer Failure Reasons 21
Development target package installation 26
6.1.2 Intel® Inspector Command line interface installation 27
6.1.3 Intel® VTune™ Amplifier Collectors Installation on Remote Systems 27
6.1.4 Preparing a Target Android* System for Remote Analysis 27
6.1.5 Intel® VTune™ Amplifier Sampling Enabling Product Installation 28
6.1.6 Intel® Integrated Performance Primitives redistributable installation 28
6.1.7 Intel® Math Kernel Library redistributable shared object installation 28
6.1.8 Intel® C++ Compiler dynamic runtime library installation 29
Eclipse* IDE Integration 29
6.1.9 Installation 29
6.1.10 Launching Eclipse for Development with the Intel C++ Compiler 29
6.1.11 Editing Compiler Cross-Build Environment Files 30
6.1.12 Cheat Sheets 30
6.1.13 SVEN Trace Viewer integration into Eclipse* 30
6.1.14 Integrating the provided GDB into Eclipse* for remote debug 30
6.1.15 Integrating the Intel® System Debugger into Eclipse* 31
Wind River* Workbench* IDE Integration 32
6.1.16 Documentation 32

6.1.17 Installation 32
6.1.18 Manual installation 33
6.1.19 Uninstall....33
Installing Intel® XDP3 JTAG Probe 34
Installing Macraigor Systems* usb2Demon* Support....34
Ordering JTAG Device for Intel® System Debugger 35
6.1.20 Intel® ITP-XDP3 35
6.1.21 Macraigor* usb2Demon*....35
Removing the Product 35
7 Issues and Limitations....36
Known Issues and Limitations 36
7.1.1 Documentation Links and FAT32 file system 36
7.1.2 Running online-installer behind proxy server fails 36
Graphics Analysis Tools 36
7.1.3 Installation failure on Windows* host with script custom actions 36
7.1.4 Support for Intel® Atom™ Processor Z3560 and Z3580 code-named “Moorefield” missing 37
Intel® Energy Profiler 37
7.1.5 \boot\config-‘uname –r’ file must be present on platform. 37
7.1.6 Power and Frequency Analysis support for Intel® Atom™ Processor covers Android* OS only. 37
Intel® VTune™ Amplifier Usage with Yocto Project* 37
7.1.7 Building Sampling Collector (SEP) for Intel® VTune™ Amplifier driver on host Linux* system 37
7.1.8 Remote Intel® VTune™ Amplifier Sampling on Intel® 64 Yocto Project* Builds ..38
7.1.9 Building 64bit Sampling Collector against Yocto Project* targeting Intel® Atom™ Processor E38xx requires additional build flags 38
Intel® System Debugger 38
7.1.10 Intel® Puma™ 6 Media Gateway Firmware Recovery Tool not available....38
7.1.11 Using the symbol browser on large data sets and large symbol info files not recommended....38
7.1.12 Limited support for Dwarf Version 4 symbol information 38

GDB* - GNU* Project Debugger 38
7.1.13 Eclipse* integration of GDB* requires Intel® C++ Compiler install....38
Software Visible Event Nexus (SVEN)....39
7.1.14 SVEN SDK is not targeting Intel® Atom™ Processor CE series 39
Intel® Integrated Performance Primitives 39
7.1.15 Some Intel® IPP domains are not installed by default....39
Intel® C++ Compiler....39
7.1.16 “libgcc_s.so.1” should be installed on the target system 39
8 Attributions....40
9 Disclaimer and Legal Information....41

1 Introduction

This document provides a brief overview of the Intel® System Studio 2015 and provides pointers to where you can find additional product information, technical support, articles and whitepapers.

It also explains how to install the Intel® System Studio product. Installation is a multi-step process and may contain components for the development host and the development target. Please read this document in its entirety before beginning and follow the steps in sequence.

The Intel® System Studio consists of multiple components for developing, debugging, tuning and deploying system and application code targeted towards embedded, Intelligent Systems, Internet of Things and mobile designs.

The tool suite covers several different use cases targeting development for embedded intelligent system platforms ranging from Intel® Atom™ Processor based low-power embedded platforms to 3rd and 4th generation Intel® Core™ microarchitecture based designs. Please refer to the Intel® System Studio User’s Guide for guidance on how to apply Intel® System Studio to the various use case scenarios that are available with this versatile product.

Due to the nature of this comprehensive integrated software development tools solution, different Intel® System Studio components may be covered by different licenses. Please see the licenses included in the distribution as well as the Disclaimer and Legal Information section of these release notes for details.

Change History

This section highlights important changes from the previous product version and changes in product updates. For information on what is new in each component, please read the individual component release notes.

Intel® System Studio 2015 Update 1

1. Support for Yocto Project* 1.7 target
2. Wind River* Linux* 7 target ready
3. Improved support for Android* 5.0 by all components
4. Intel® Threading Building Blocks Android* 5.0 Lollipop support
5. 64-bit Android* support by Intel® C++ Compiler, Intel® Integrated Performance Primitives and Intel® Threading Building Blocks
6. Intel® Math Kernel Library Optimizations for Intel® Advanced Vector Extensions 512 (Intel® AVX-512)
7. Intel® System Debugger improved support for the Intel® Atom™ Processor N29xx, E38xx, Z374x and Z377x code-named “Baytrail”
8. Intel® System Debugger Improved support for Intel® Quark™ SoC with OpenOCD*

Intel® System Studio 2015

1. Intel® Edison module support by Intel® C++ Compiler, Intel® Integrated Performance Libraries, Intel® Math Kernel Library, Intel® Threading Building Blocks and GDB
2. Wind River* Workbench* IDE integration and platform recipes for Intel® C++ Compiler, Intel® Integrated Performance Primitives, Intel® Math Kernel Library, Intel® VTune™ Amplifier Sampling Collector
3. Wind River* Workbench* IDE launcher for Intel® VTune™ Amplifier for Systems and Intel® System Debugger
4. Run-control and reset handling improvements for OpenOCD* based debug with the Intel® System Debugger
5. Intel® Integrated Performance Primitives 8.2
6. Enhanced Intel® Quark SoC support with Intel® C++ Compiler, Intel® Integrated Performance Primitives, Intel® System Debugger
6. Intel® VTune™ Amplifier for Systems
   6.1. Intel® VTune Amplifier for Systems enhanced profiling methods (driverless solutions via Perf*, F-Trace*)
   6.2. GPU profiling for Android*
   6.3. Eclipse* plug-in
   6.4. OS X* host support
7. Intel® Inspector for Systems:
   7.1. Deeper analysis method to substantially reduce the number of false positives.
   7.2. Independent control of uninitialized memory analysis
   7.3. Improved on-demand leak detection and memory growth interface.
   7.4. Performance improvement in threading error analysis.
8. Intel® System Debugger
   8.1. Intel® System Debugger launch plug-ins for Eclipse* and Wind River Workbench*

8.2. Instruction trace window and source level instruction flow visualization improvements
8.3. Intel® System Debugger using OpenOCD communication protocol for Intel® Quark SoC targets with low-cost JTAG devices
9. Intel® C++ Compiler
9.1. Unified Intel® C++ Compiler 15.0 for both Embedded Linux* and Android* targets,
10. Support for native code generation for Intel® Graphics Technology
11. Intel® Math Kernel Library
1.1. Intel® Math Kernel Library 11.2 featuring Intel® Atom™ processor support
12. Updated and redesigned embedded use case how-to guides and documentation

Intel® System Studio 2014 Update 2

1. Intel® VTune™ Amplifier
1.1. Sampling collector improvements for Android* target
2. Intel® System Debugger
2.1. Support for Intel® Atom™ Processor C2xxx, E3xxx, Z32xx, Z33xx as well as 4th generation Intel® Core™ processors with Linux* hosted debugger.
2.2. Support of BIOS flashing for 4th generation Intel® Core™ processor
2.3. Support for 3rd generation Intel® Core™ processor target
2.4. Support for latest generation Intel® Atom™ Processor Z3460 and Z3480.

Intel® System Studio 2014 Update 1

1. Intel® IPP support of the Intel® Quark SoC has been added for a subset of the libraries
2. Intel® C++ Compiler for Android* and embedded OS Linux* updated
3. Intel® VTune™ Amplifier
3.1. Updated version of Intel® VTune™ Amplifier with support for remote software-based algorithm analysis (Basic Hotspots, Concurrency, Locks and Waits) on embedded Linux target systems
3.2. New analysis type “TSX Exploration” for 4th generation Intel® Core™ processors.
3.3. Support for external data collection launched from the VTune Amplifier with the Custom collector target configuration option or -custom-collector command line option.
3.4. Android 64 bit kernel support (not 64-bit user space)

Intel® System Studio 2014

1. Operating System Support
1.1. Windows* 7 and 8 host support for Linux* targeted development
1.2. Android 4.0.x through 4.4.x target support
1.3. Yocto Project* 1.2 through 1.5 target support
1.4. Wind River* Linux* 5 native build support
1.5. Tizen* IVI 2.x and 3.x target support
1.6. Graphics Performance Analyzer for Android* target now included.
1.7. Integration into Eclipse* build environment on both Linux* and Windows* OS
2. Platform Support
2.1. Support for 4th generation Intel® Core™ processor
2.2. Support for Intel® Atom Processor Z3xxx, E3xxx, C2xxx
2.3. Support for Intel® Xeon® E5-2600 v2 & E5-1600 v2 processors
2.4. Intel® C++ Compiler support for Intel® Quark processor

2.5. Intel® C++ Compiler optimizations for next generation Intel® Atom™ processor
2.6. Intel® System Debugger support for 4th generation Intel® Core™ processor and Intel® Xeon™ processor as well as next generation Intel® Atom™ processor.

3. Component Features
3.1. System-wide Intel® VTune™ Amplifier memory bandwidth and performance analysis support
3.2. Improved sysroot cross-build integration support

Intel® Software Manager

The installation now provides an Intel® Software Manager to provide a simplified delivery mechanism for product updates and provide current license status and news on all installed Intel software products.

You can also volunteer to provide Intel anonymous usage information about these products to help guide future product design. This option, the Intel® Software Improvement Program, is not enabled by default – you can opt-in during installation or at a later time, and may opt-out at any time. For more information please see http:\\intel.ly\SoftwareImprovementProgram.

Product Contents

The product contains the following components

1. Intel® C++ Compiler 15.0
2. Intel® Integrated Performance Primitives 8.2 for Linux*
3. Intel® Math Kernel Library 11.2 for Linux*
4. Intel® Threading Building Blocks 4.3
5. Intel® System Debugger 2015
    5.1. Intel® System Debugger notification module xdbntf.ko (Provided under GNU General Public License v2)
6. OpenOCD 0.8.0 library (Provided under GNU General Public License v2+)
    6.1. OpenOCD 0.8.0 source (Provided under GNU General Public License v2+)
7. GNU* GDB 7.7 (Provided under GNU General Public License v3)
    7.1. Source of GNU* GDB 7.7 (Provided under GNU General Public License v3)
8. SVEN Technology 1.0 (SDK provided under GNU General Public License v2)
9. Intel® VTune™ Amplifier 2015 for Systems
    9.1. Intel® VTune™ Amplifier Sampling Enabling Product (SEP) 3.15
    9.2. Intel® Energy Profiler
    9.3. WakeUp Watch for Android* 3.1.6
    9.4. SoC Watch for Android* 1.5
10. Intel® Inspector 2015 for Systems
11. Intel® System Studio System Analyzer 2014 R3

2 Getting Started

Please refer to the Getting Started Guide and Intel® System Studio User's Guide for guidance on Intel® System Studio use cases and supported usage models.

Intel® System Studio User's Guide

- `<install-dir>System Studio 2015.1.xxx \Documentation\en_US\embedded_compsupdoc_l.pdf`

Intel® System Studio Getting Started Guide

- `<install-dir>\System Studio 2015.1.xxx\Documentation\en_US\embedded_gsdoc_w.htm`

3 Technical Support and Documentation

Release Notes Location

The release notes and getting started guide for the tools components making up the Intel® System Studio product can be found at the following locations after installation is complete.

The paths are given relative to the installation directory <install-dir>. The default installation directory is C:\Program Files (x86)\Intel unless indicated differently.

Intel® System Studio Release Notes and Installation Guide

- `<install-dir>\System Studio 2015.1.xxx\Documentation\en_US\all-release-install.pdf`

Intel® C++ Compiler

- `<install-dir>\System Studio 2015.1.xxx\Documentation\en_US\compiler_c\ReleaseNotes_ISS_Compiler.pdf`

Intel® Integrated Performance Primitives

- `<install-dir>\System Studio 2015.1.xxx\Documentation\en_US\ipp\ReleaseNotes.htm`

Intel® Math Kernel Library

- `<install-dir>\System Studio 2015.1.xxx\Documentation\en_US\mkl\ReleaseNotes.htm`

Intel® System Debugger

- `<install-dir>\System Studio 2015.1.xxx\Documentation\en_US\debugger\xdb\sysdebug-release-install.pdf`

SVEN SDK

- `<install-dir>\System Studio 2015.1.xxx\Documentation\en_US\debugger\sven\sdk\SVEN_SDK.pdf`

SVEN Trace Viewer

- `<install-dir>\System Studio 2015.1.xxx\Documentation\en_US\debugger\sven\viewer\SVEN_VIEWER.pdf`

GDB

- `<install-dir>\system_studio_2015.1.xxx\Documentation\en_US\debugger\gdb\GDB_Release_notes.pdf`

- `<install-dir>\system_studio_2015.1.xxx\Documentation\en_US\debugger\gdb\GDB.pdf`

Intel® VTune™ Amplifier

- `<install-dir>\System Studio 2015.1.xxx\Vtune Amplifier 2015 for Systems\Documentation\en\release_notes_amplifier_for_systems_windows.pdf`

Intel® Inspector

- `<install-dir>\System Studio 2015.1.xxx\Inspector 2015 for Systems\Documentation\en\Release_Notes_Inspector_Windows.pdf`

Intel® VTune™ Amplifier Sampling Enabling Product

- `<install-dir>\System Studio 2015.1.xxx\Vtune Amplifier 2015 for Systems\Documentation\en\SEP_Users_Guide.pdf`

WakeUp Watch for Android*

- The user's guide explaining usage can be found at `../system_studio_target/wuwatch_android/WakeUpWatchForAndroid.pdf` after unpacking the `<install-dir>\targets\system_studio_target.tgz` package.

SoC Watch for Android*

- The user's guide explaining usage can be found at `../system_studio_target /socwatch_android/SoCWatchForAndroid_v1_4_0.pdf` after unpacking the `<install-dir>\targets\system_studio_target.tgz` package.

Intel® System Studio System Analyzer

- http://software.intel.com/en-us/articles/intel-graphics-performance-analyzers-for-android-os
- http://software.intel.com/en-us/articles/intel-gpa-online-help

Article & Whitepaper Locations

Intel® System Studio Tutorials and Samples

- `<install-dir>\System Studio 2015.1.xxx\Documentation\en_US\samples-and-tutorials.html`

Intel® System Studio Articles and Whitepapers

- https://software.intel.com/en-us/articles/intel-system-studio-articles
- For a list of all available articles, whitepapers and related resources please visit the Intel® System Studio product page at http:\\software.intel.com\en-us\intel-system-studio and look at the Support tab.

Support

If you did not register your compiler during installation, please do so at the Intel® Software Development Products Registration Center. Registration entitles you to free technical support, product updates and upgrades for the duration of the support term.

To submit issues related to this product please visit the Intel Premier Support webpage and submit issues under the product **Intel(R) System Studio**.

Additionally you may submit questions and browse issues in the Intel® System Studio User Forum.
For information about how to find Technical Support, product documentation and samples, please visit http:\\software.intel.com\en-us\intel-system-studio.

Note: If your distributor provides technical support for this product, please contact them for support rather than Intel.

Optimization Notice
Intel's compilers may or may not optimize to the same degree for non-Intel microprocessors for optimizations that are not unique to Intel microprocessors. These optimizations include SSE2, SSE3, and SSSE3 instruction sets and other optimizations. Intel does not guarantee the availability, functionality, or effectiveness of any optimizations on microprocessors not manufactured by Intel. Microprocessor-dependent optimizations in this product are intended for use with Intel microprocessors. Certain optimizations not specific to Intel microarchitecture are reserved for Intel microprocessors. Please refer to the applicable User and Reference Guides for more information regarding the specific instruction sets covered by this notice. Notice revision #20110804

4 What's New

Update 1

1. Support for Yocto Project* 1.7 target
2. Wind River* Linux* 7 target ready
3. Improved support for Android* 5.0 by all components
4. Intel® Threading Building Blocks Android* 5.0 Lollipop support
5. 64-bit Android* support by Intel® C++ Compiler, Intel® Integrated Performance Primitives and Intel® Threading Building Blocks
6. Intel® Math Kernel Library Optimizations for Intel® Advanced Vector Extensions 512 (Intel® AVX-512)
7. Intel® System Debugger improved support for the Intel® Atom™ Processor N29xx, E38xx, Z374x and Z377x code-named “Baytrail”
8. Intel® System Debugger Improved support for Intel® Quark™ SoC with OpenOCD*

Initial Release

The Intel® System Studio has a set of new features specifically targeting cross-development needs and also adds support for new Intel® Architecture based hardware platforms:

1. Intel® Edison module support by Intel® C++ Compiler, Intel® Integrated Performance Libraries, Intel® Math Kernel Library, Intel® Threading Building Blocks and GDB. Please contact us at IntelSystemStudio@intel.com for Intel® Edison module support by other components.
2. Enhanced Intel® Quark SoC support with Intel® C++ Compiler, Intel® Integrated Performance Primitives, Intel® System Debugger
3. Intel® VTune™ Amplifier for Systems
   3.1. Intel® VTune Amplifier for Systems enhanced profiling methods (driverless solutions via Perf*, F-Trace*)
   3.2. GPU profiling for Android*
   3.3. Eclipse* plug-in
   3.4. OS X* host support
4. Intel® Inspector for Systems:
   4.1. Deeper analysis method to substantially reduce the number of false positives.
   4.2. Independent control of uninitialized memory analysis
   4.3. Improved on-demand leak detection and memory growth interface.
   4.4. Performance improvement in threading error analysis.
5. Intel® System Debugger
   5.1. Intel® System Debugger launch plug-ins for Eclipse* and Wind River Workbench*
   5.2. Instruction trace window and source level instruction flow visualization improvements
   5.3. Intel® System Debugger using OpenOCD communication protocol for Intel® Quark SoC targets with low-cost JTAG devices
6. Intel® C++ Compiler

6.1. Unified Intel® C++ Compiler 15.0 for both Embedded Linux* and Android* targets,
7. Intel® Math Kernel Library
   7.1. Intel® Math Kernel Library 11.2 featuring Intel® Atom™ processor support
8. Updated and redesigned embedded use case how-to guides and documentation
9. Improved Wind River* Workbench* and Eclipse IDE* integration
10. Intel® Threading Building Blocks 4.3

5 System Requirements

Supported Host Platforms

One of the following operation distributions (this is the list of distributions supported by all components; other distributions may or may not work and are not recommended - please refer to Technical Support if you have questions).

- Windows* 7, 8.x

If you have any doubts about installation requirements, please check the Prerequisites Guide

Individual Intel® System Studio 2015 components may support additional distributions. See the individual component's installation guide and release notes after you ran the installer for the tool suite distribution

```
w_cembd_p_2015.1.xxx.exe
```

or

```
w_cembd_p_2015.1.xxx_online.exe
```

for details.

Eclipse* Integration Prerequisites

When asked point the installer to the installation directory of your Eclipse* install. Usually this would be `C:\Program Files (x86)\eclipse\.`

The prerequisites for successful Eclipse integration are:

1. Eclipse* 3.8 (Juno) – Eclipse* 4.4 (Kepler)
2. Eclipse* CDT 8.0 – 8.4
3. Java Runtime Environment (JRE) version 6.0 (also called 1.6) update 11 or later.

Host Prerequisites and Resource Requirements

5.1.1 Host Space Requirements by Component

	Minimum RAM	Recommended RAM	Disk Space
Intel® System Studio	2Gb	4Gb	7Gb
Intel® C++ Compiler	1Gb	2Gb	2.5Gb
Intel® Integrated Performance Primitives	1Gb	4Gb	1-2Gb
Intel® Math Kernel Library	1Gb	4Gb	2.3Gb
Intel® VTune™ Amplifier for Systems	2Gb	4Gb	650Mb
Intel® Inspector for Systems	2Gb	4Gb	350Mb
GDB	1Gb	2Gb	200Mb
Intel® System Debugger	1Gb	2Gb	300Mb
SVEN	2Gb	4Gb	350Mb

5.1.2 Intel® Integrated Performance Primitives (Intel® IPP) Details

Intel® Integrated Performance Primitives (Intel® IPP) for IA-32 Hardware Requirements:

- 1800MB of free hard disk space, plus an additional 400MB during installation for download and temporary files.

Intel® Integrated performance Primitives (Intel® IPP) for Intel® 64 Hardware Requirements:

- 1900MB of free hard disk space, plus an additional 700MB during installation for download and temporary files.

5.1.3 Intel® C++ Compiler

Cross-build for Wind River Linux* target currently requires an existing Wind River* Linux 4.x, 5.x or 6.x installation that the compiler can integrate into.

Target Software Requirements

The target platform should be based on one of the following environments:

- Yocto Project* 1.4, 1.5, 1.6, 1.7 based environment
- CE Linux* PR35 based environment
- Tizen* IVI 3.x
- Wind River* Linux* 4, 5, 6 based environment
- Android* 4.0.x through 4.4.x

Note:

The level of target OS support by a specific Intel® System Studio component may vary.

Target Prerequisites and Resource Requirements

5.1.4 Target Space Requirement by Component

	Minimum RAM	Dependencies	Disk Space
Intel® C++ Compiler	application dependent	Linux kernel 1.26.18 or newer glibs-2.5 or compatible libgcc-4.1.2 or compatible libstdc++-3.4.7 or compatible	13Mb (IA-32) 15Mb (Intel® 64)
Intel® VTune™ Amplifier CLI	4Gb	Specific kernel configuration reqs. Details below.	200Mb
Intel® VTune™ Amplifier SEP	(# logical cores+2) Mb	Specific kernel configuration reqs. Details below.	8Mb
SoC Watch	(# logical cores+2) Mb	Specific kernel configuration reqs. See SoCWatch documentation	8Mb
WakeUp Watch	(# logical cores+2) Mb	Specific kernel configuration reqs. See WuWatch documentation	8Mb
Intel® Inspector for Systems CLI	2Gb	4Gb	350Mb
gdbserver	negligable	none	1.5Mb
xdbntf.ko	<1Mb	kernel build environment	<1Mb
SVEN	<1Mb	kernel build environment	sampling dependent

5.1.5 Intel® VTune™ Amplifier target OS kernel configuration

For Intel® VTune™ Amplifier performance analysis and Intel® Energy Profiler there are minimum kernel configuration requirements. The settings below are required for different analysis features.

- For event-based sampling (EBS) sep3_x.ko and pax.ko require the following settings:
  CONFIG_PROFILING=y
  CONFIG_OPROFILE=m (or CONFIG_OPROFILE=y)
  CONFIG_HAVE_OPROFILE=y

- For EBS with callstack information vtsspp.ko additionally needs the following settings:
  CONFIG_MODULES=y
  CONFIG_SMP=y
  CONFIG_MODULE_UNLOAD=y
  CONFIG_KPROBES=y
  RING_BUFFER=y
  CONFIG_TRACEPOINTS=y (optional but recommended)

- For power analysis, required by apwr3_x.ko
  CONFIG_MODULES=y
  CONFIG_MODULE_UNLOAD=y
  CONFIG_TRACEPOINTS=y
  CONFIG_FRAME_POINTER=y
  CONFIG_COMPAT=y
  CONFIG_TIMER_STATS=y
  CONFIG_X86_ACPI_CPUFREQ=m (or CONFIG_X86_ACPI_CPUFREQ=y)
  CONFIG_INTEL_IDLE=y

5.1.6 Intel® VTune™ Amplifier Feature vs. Resource Matrix

	Event based sampling (EBS) analysis	EBS analysis with stacks	Algorithmic analysis (PIN-based)	Intel Energy Profiler	Remote collection from host	Result view on target	Requirements:
SEP "VTune Amplifier hardware event-based sampling collector for performance analysis"	X						~8 MB disk space (Number of logical cores +2) Mb RAM
amplxe-cl -target "VTune Amplifier collector for power and performance analysis on Embedded Linux systems"	X	X	X	X	X		~25 MB disk space ~64 Mb RAM
amplxe-cl "VTune Amplifier command line interface for text-based power and performance analysis"	X	X	X	X	X	X	~200MB disk space >= 4Gb RAM

Hardware Requirements

- IA32 or Intel® 64 architecture based host computer
- Development platform based on the Intel® Atom™ processor Z5xx, N4xx, N5xx, D5xx, E6xx, N2xxx, D2xxx, Z2xxx, Z3xxx, E3xxx, C2xxx or Intel® Atom™ processor CE4xxx, CE53xx and the Intel® Puma™ 6 Media Gateway
- Intel® Edison development platform
- Alternatively development platform based on 2nd , 3rd or 4th generation Intel® Core™ processor.
- Alternatively development platform based on 2nd , 3rd or 4th generation Intel® Xeon® processor.
- Alternatively development platform based on 5th generation Intel® Core™ M processor

6 Installation Notes

Installing the Tool Suite

The default installation directories are:

- `C:\Program Files (x86)\Intel\System Studio 2015.1.xxx\`
- `C:\Program Files (x86)\Intel\System Studio 2015.1.xxx\bin`
- `C:\Program Files (x86)\Intel\System Studio 2015.1.xxx\compiler`
- `C:\Program Files (x86)\Intel\System Studio 2015.1.xxx\ipp`
- `C:\Program Files (x86)\Intel\System Studio 2015.1.xxx\mkl`
- `C:\Program Files (x86)\Intel\System Studio 2015.1.xxx\tbb`
- `C:\Program Files (x86)\Intel\System Studio 2015.1.xxx\debugger\xdb`
- `C:\Program Files (x86)\Intel\System Studio 2015.1.xxx\debugger\gdb`
- `C:\Program Files (x86)\Intel\System Studio 2015.1.xxx\debugger\sven`
- `C:\Program Files (x86)\Intel\System Studio 2015.1.xxx\vtune_amplifier_2015_for_systems\`
- `C:\Program Files (x86)\Intel\System Studio 2015.1.xxx\inspector_2015_for_systems\`

for Intel® C++ Compiler, Intel® Integrated Performance Primitives, Intel® Math Kernel Library, Intel® Threading Building Blocks, Intel® System Debugger, Intel® VTune™ Amplifier, and Intel® Inspector respectively.

The Intel® System Studio contains components under GNU* Public License (GPL) in addition to commercially licensed components. This includes the GNU* Project Debugger – GDB, the SVEN SDK and the kernel module used by the Intel® System Debugger to export Linux* dynamically kernel module memory load information to host.

The Intel® VTune™ Amplifier, Intel® Energy Profiler and Intel® Inspector are available for power and performance tuning as well as memory and thread checking on the installation host.

For additional installation of command-line only versions of Intel® VTune™ Amplifier, SoC Watch, WakeUp Watch and Intel® Inspector on the development target, please follow the sub-chapter on the command line interface (CLI) installations below.

Furthermore a target package `system_studio_target.tgz` in `C:\Program Files (x86)\Intel\System Studio 2015.1.xxx\Targets` contains Intel® C++ Compiler runtime libraries, the Intel® VTune™ Amplifier Sampling Enabling Product (SEP) , target

components for the Intel® VTune™ Amplifier Data Collector and the kernel module used by the Intel® System Debugger to export Linux* dynamically kernel module memory load information to host.

The Intel® System Debugger at this point is intended for Intel® Atom™ processor targeted system software cross-debug.

Sudo or Root Access Right Requirements

- Integration of the Intel® C++ Compiler into the Yocto Project* Application Development Toolkit requires the launch of the tool suite installation script install.sh as root or sudo user.
- Installation of the hardware drivers for the Intel® ITP-XDP3 probe to be used with the Intel® System Debugger requires the launch of the tool suite installation script install.sh as root or sudo user.

Using the online installer

For installation of the tool suite on the development host please follow the steps below:

1. Ensure that you are connected to the internet and that https protocol based component downloads are permitted by your firewall.
2. Execute the online installer executable.
   `> w_cembd_2015.1.xxx_online.exe`
3. After launching the online installer you will see the Installer splash screen with a small message stating: "`download_configuration_files`". Afterwards the installation follows the same steps as the full product install, except that the selected components are being downloaded from the server one by one.
4. The target package system_studio_target.tgz will be located in `C:\Program Files (x86)\Intel\System Studio 2015.1.xxx\Targets` after the host installation is complete.
5. Follow the same steps as outlined in "Installing the full distribution package" steps 5 through 18 below.

6.1.1 Online Installer Failure Reasons

6.1.1.1 Symptom 1: Online install bootstrapper cannot connect to IRC.

```
Starting installer...
This is an online-installer for Intel(R) System Studio 2015
Connection to the IRC site cannot be established.
```

Root Cause:

- Port 443 not opened in firewall
- Port 443 not opened in ACL for specific server (can be verified using network trace).
- The Intel® System Studio online installer currently does not fully support proxy servers.

6.1.1.2 Symptom 2: Download stops in an infinite loop

Download stops in an infinite loop.

```
Extracting data...
Starting installer...
```

Root Cause:

- Proxy server not specified on calling shell

For installation of the tool suite on the development host please follow the steps below:

1. Upon registering for the program you will receive a serial number and email with a license file. You will need either of these two to complete the installation process. If you want to use the license file you can point to it during install, but you can also copy it to `C:\Program Files (x86)\Common Files\Intel\Licenses\`for automatic pickup by the installer.
2. Execute one of the installer executables.

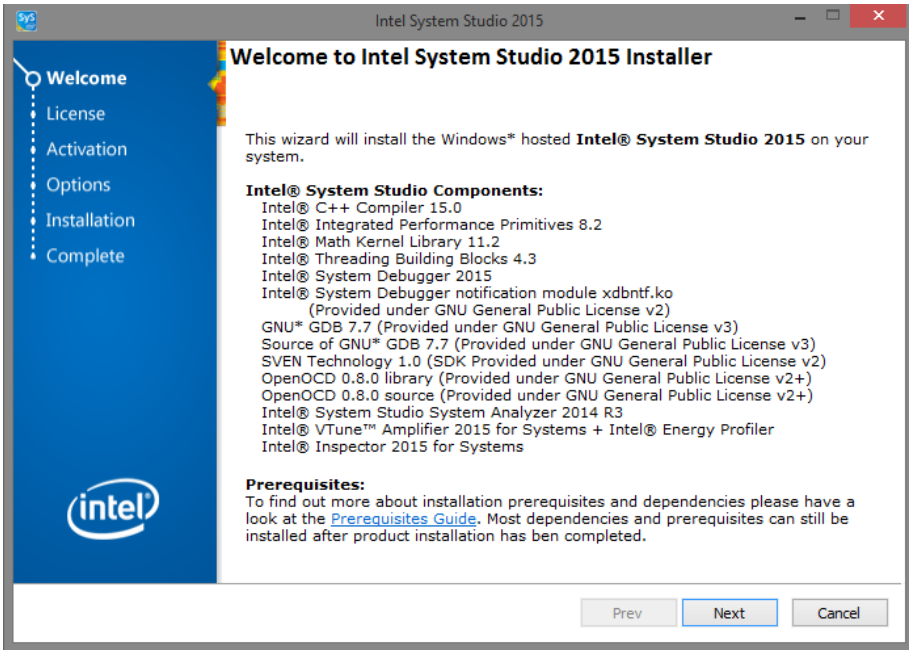
   `w_cembd_2015.1.xxx.exe`

   `or`

   `w_cembd_2015.1.xxx_online.exe`

   The later one is an online installer reducing the initial package download size

3. The welcome message to the Intel® System Studio installation process appears.

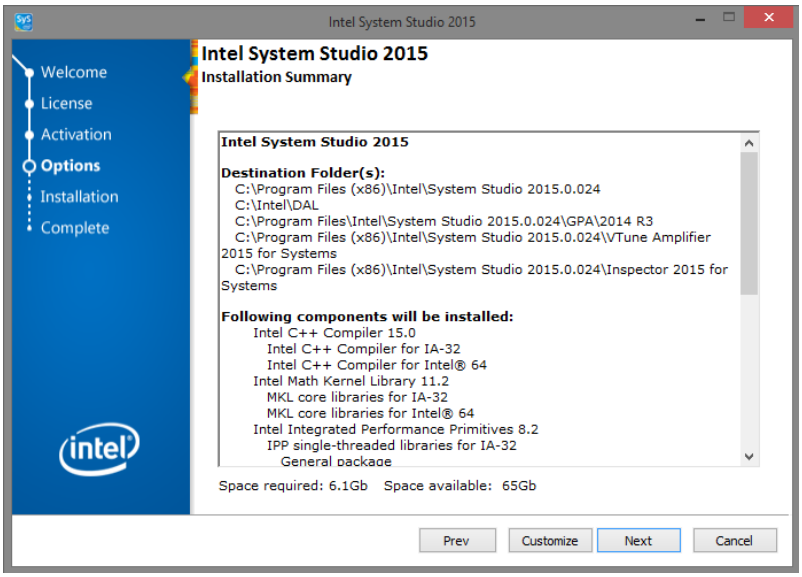

Intel System Studio 2015
Welcome
License
Activation
Options
Installation
Complete
Welcome to Intel System Studio 2015 Installer
This wizard will install the Windows* hosted Intel® System Studio 2015 on your system.
Intel® System Studio Components:
Intel® C++ Compiler 15.0
Intel® Integrated Performance Primitives 8.2
Intel® Math Kernel Library 11.2
Intel® Threading Building Blocks 4.3
Intel® System Debugger 2015
Intel® System Debugger notification module xdbntf.ko
(Provided under GNU General Public License v2)
GNU* GDB 7.7 (Provided under GNU General Public License v3)
Source of GNU* GDB 7.7 (Provided under GNU General Public License v3)
SVEN Technology 1.0 (SDK Provided under GNU General Public License v2)
OpenOCD 0.8.0 library (Provided under GNU General Public License v2+)
OpenOCD 0.8.0 source (Provided under GNU General Public License v2+)
Intel® System Studio System Analyzer 2014 R3
Intel® VTune™ Amplifier 2015 for Systems + Intel® Energy Profiler
Intel® Inspector 2015 for Systems
Prerequisites:
To find out more about installation prerequisites and dependencies please have a look at the Prerequisites Guide. Most dependencies and prerequisites can still be installed after product installation has ben completed.
Prev
Next
Cancel

   Selecting `Next` will move you to the next step in the installation process.

   If you have any doubts about installation requirements, please check the Prerequisites Guide.

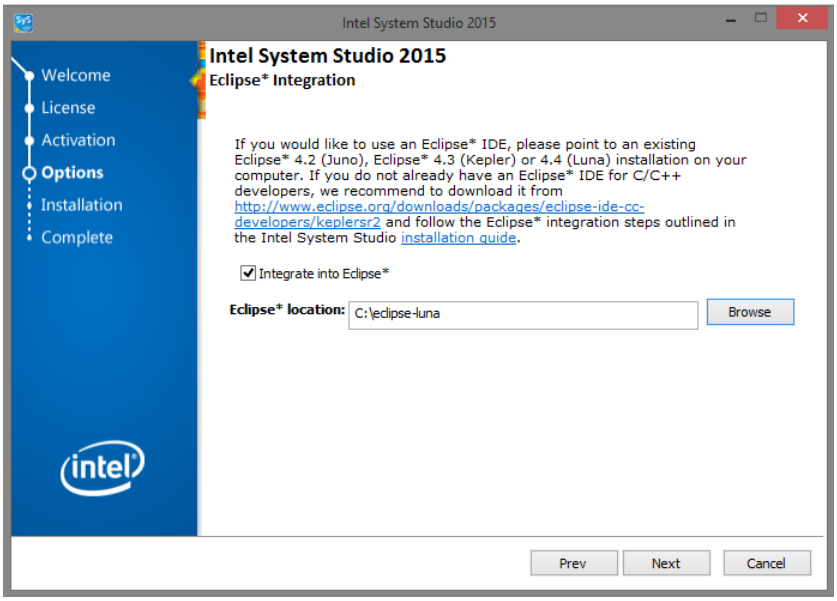
4. The installation routine checks for the availability of all product dependencies. Please address these dependencies, if a warning message appears.
5. Afterwards you will be asked to read the end-user license agreement for the tool suite.
6. When asked whether you would like to activate and install your product select one of the options provided depending on whether you have a license file available or not. If there is already a valid license file available and installed on your system, the installation routine will recommend to simply use the existing license file. If you do not have access to the internet at the time of installation, select the alternative activation option.
7. The next screen will ask you whether you would like to participate in the Intel® Software Improvement Program. This will help us identify opportunities for product improvement.
8. The following screen lets you review your installation options.

9. If you would like to only install a specific tool suite component, and change the components settings you can select custom installation.

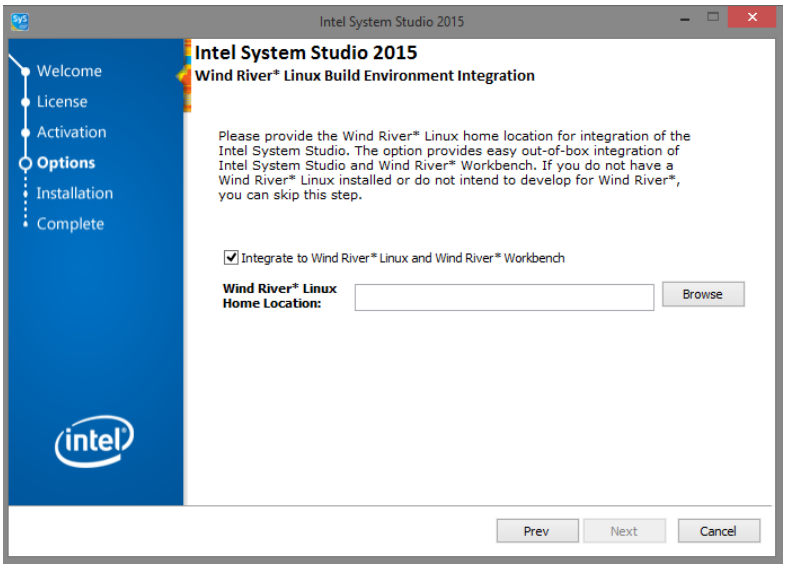


10. Select the installation folder.
11. During the Intel® System Studio 2015 installation you will see installation messages for the components you selected.
12. In the next step you can decide to have the installation program do the Eclipse* CDT integration of Intel® System Studio for you. You can select the installation directory for your existing Eclipse* installation and have the Intel® System Studio installer automatically integrate
    - Intel® C++ Compiler
    - SVEN Trace Viewer
    - Intel's GDB distribution
    - Intel® System Debugger
    - Intel® VTune™ Amplifier for Systems

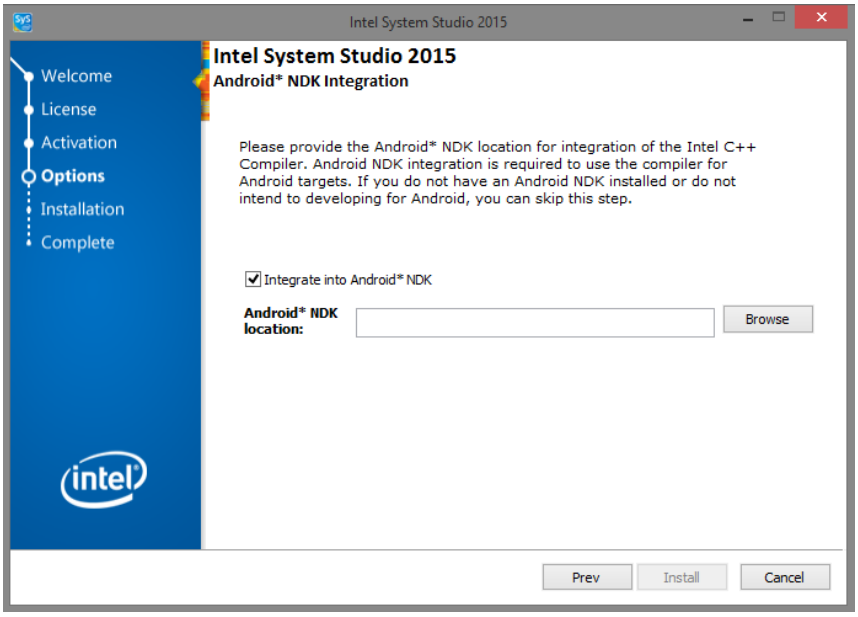
into Eclipse* for you.


Intel System Studio 2015
Welcome
License
Activation
Options
Installation
Complete
Intel System Studio 2015
Eclipse* Integration
If you would like to use an Eclipse* IDE, please point to an existing Eclipse* 4.2 (Juno), Eclipse* 4.3 (Kepler) or 4.4 (Luna) installation on your computer. If you do not already have an Eclipse* IDE for C/C++ developers, we recommend to download it from http://www.eclipse.org/downloads/packages/eclipse-ide-cc-developers/keplersr2 and follow the Eclipse* integration steps outlined in the Intel System Studio installation guide.
Integrate into Eclipse*
Eclipse* location:
C:\eclipse-luna
Browse
Prev
Next
Cancel

13. Afterwards you can also choose to integrate the following components into Wind River* Linux* build environment:
    - Intel® C++ Compiler
    - Intel® Integrated Performance Primitives
    - Intel® Math Kernel Library
    - Intel® System Debugger
    - Intel® VTune™ Amplifier for Systems


Intel System Studio 2015
Welcome
License
Activation
Options
Installation
Complete
Intel System Studio 2015
Wind River* Linux Build Environment Integration
Please provide the Wind River* Linux home location for integration of the Intel System Studio. The option provides easy out-of-box integration of Intel System Studio and Wind River* Workbench. If you do not have a Wind River* Linux installed or do not intend to develop for Wind River*, you can skip this step.
Integrate to Wind River* Linux and Wind River* Workbench
Wind River* Linux Home Location:
Browse
Prev
Next
Cancel

14. Finally you can have the Intel® C++ Compiler automatically integrated with an Android* NDK present on your system.


Intel System Studio 2015
Welcome
License
Activation
Options
Installation
Complete
Intel System Studio 2015
Android* NDK Integration
Please provide the Android* NDK location for integration of the Intel C++ Compiler. Android NDK integration is required to use the compiler for Android targets. If you do not have an Android NDK installed or do not intend to developing for Android, you can skip this step.
Integrate into Android* NDK
Android* NDK location:
Browse
Prev
Install
Cancel

15. After the installation is complete an installation summary screen will be displayed and the Intel® System Studio Getting Started Guide will be automatically opened.

    Note: Your browser may block Java Script* on the page by default (depending on browser settings) and ask the user to enable it. If the user does not enable it, some links on the page will not work.

16. The various Intel® System Studio components that are available outside of Eclipse* can be accessed through the Windows* OS start menu.

Development target package installation

The targets directory contains Intel® C++ Compiler runtime libraries, the Intel® VTune™ Amplifier Sampling Enabling Product (SEP) , target components for the Intel® VTune™ Amplifier Data Collector, target components for the Intel® Inspector, the xdbntf.ko used by the Intel® System Debugger to export Linux* kernel module memory load information to host, and prebuilt gdbserver target debug agents for GDB.

To install it follow the steps below

1. Copy the contents of the `C:\Program Files (x86)\Intel\System Studio 2015.1.xxx\targets` directory to your target platform and unpack unpack the system_studio_target.tgz and debugger_kernel_module.tgz files contained in this directory there.

2. Add the compiler runtime libraries that you find in `../system_studio_target/compiler/lib/ia32` to your target environment search path.
3. For the dynamic kernel module load export feature follow the instructions found at `../debugger_kernel_module/debugger/xdb/kernel-modules/xdbntf/read.me`.
   This is also detailed in the Intel® System Debugger Installation Guide and Release Notes `jtag-release-install.pdf`.
4. For the GDB* Debugger remote debug agent gdbserver pick the executable that describes your target system from `../<arch>/<target>/bin`, where `arch` and `target` can have the following values
   - `arch: ia32, intel64, Quark`
   - `target: WindRiverLinux5, WindRiverLinux4, TizenIVI, CELinuxPR32, Yocto1.3, Yocto1.4, Yocto1.5, Galileo`

   Run gdbserver on the target platform to enable remote application debug.
   During the Intel® System Studio product install you can also choose to install the gdbserver sources if support for additional target platforms is needed.
5. For the Intel® VTune Amplifier Sampling Enabling Product (SEP) pick `../system_studio_target/vtune_amplifier_2015_for_systems_target/linux<arch>/vtune_amplifier_target_sep_x86[_64].tgz`
   - `arch: 32, 64`
6. For the Intel® VTune Amplifier for Systems target package pick `../system_studio_target/vtune_amplifier_2015_for_systems_target/linux<arch>/vtune_amplifier_target_x86[_64].tgz`
   - `arch: 32, 64`
7. For WakeUp Watch for Android* follow the instructions at `../system_studio_target/wuwatch_android/WakeUpWatchForAndroid.pdf`
8. For SoC Watch for Android* follow the instructions at

../system_studio_target/socwatch_android_v1.4d/SoCWatchForAndroid_v1_4_0.pdf

9. For the Intel® Inspector for Systems follow the instructions in ../system_studio_target/inspector_2015_for_systems/documentation/en/Release_Notes_Inspector_Linux.pdf

6.1.2 Intel® Inspector Command line interface installation

If you would like to install the Intel® Inspector command line interface only for thread checking and memory checking on a development target device, please follow the steps outlined below:

1. From ../inspector_2015_for_systems/ on the target execute the environment configuration script inspxe-genvars.sh.
2. The fully functional command-line Intel® Inspector installation can be found in the bin32 and bin64 subdirectories for IA32 and Intel® 64 targets respectively.

6.1.3 Intel® VTune™ Amplifier Collectors Installation on Remote Systems

If you would like to install the Intel® VTune™ Amplifier data collector for power tuning and performance tuning on a development target device, please follow the steps outlined below:

1. You will find the Intel® VTune™ Amplifier data collectors at

   ../system_studio_target/vtune_amplifier_2015_for_systems_target/linux<arch>/vtune_amplifier_target_x86[_64].tgz

   arch: 32, 64

   on the target.

2. Data collection on both IA32 and Intel® 64 targets is supported.

3. Follow the instructions in Help document in section “User’s guide->Running analysis remotely” for more details, on how to use this utility.

6.1.4 Preparing a Target Android* System for Remote Analysis

If you would like to install the Intel® VTune™ Amplifier data collectors for power tuning and performance tuning on an Android* target device, please follow the steps outlined below:

4. You will find SoC Watch at

   ../system_studio_target/socwatch_android_v1.5/
   on the target.

   - ../system_studio_target/socwatch_android_v1.5/WakeUpWatchForAndroid.pdf

- The "Preparing a Target Android* System for Remote Analysis" chapter of the Intel® VTune™ Amplifier User's Guide.

5. You will find WakeUp Watch at

   `../system_studio_target/wuwatch_android`

   on the target.

   Please follow the instructions for installation and usage in

   - `../system_studio_target/socwatch_android/SoCWatchForAndroid_v1_4_0.pdf`
   - The "Preparing a Target Android* System for Remote Analysis" chapter of the Intel® VTune™ Amplifier User's Guide.

6.1.5 Intel® VTune™ Amplifier Sampling Enabling Product Installation

If you would like to install the Intel® VTune™ Amplifier Sampling Enabling Product (SEP), please follow the steps outlined below:

1. You will find the Intel® VTune Amplifier Sampling Enabling Product at
   `../system_studio_target/vtune_amplifier_2015_for_systems_target/linux<arch>/vtune_amplifier_target_sep_x86[_64].tgz`

2. After unpacking this zip file follow the instructions in
   `../vtune_amplifier_2015_for_systems.0.x.xxxxxx\sepdk\src\README.txt`

6.1.6 Intel® Integrated Performance Primitives redistributable installation

If you are using dynamic linking when using the Intel® Integrated Performance Primitives, you will need to copy the relevant Linux* shared objects to the target device along with the application.
A list of the redistributable shared objects can be found at

```
C:\Program Files (x86)\Intel\System Studio 2015.1.036\
Documentation\en_US\ipp
```

6.1.7 Intel® Math Kernel Library redistributable shared object installation

If you are using dynamic linking when using the Intel® Math Kernel Libraries, you will need to copy the relevant Linux* shared objects to the target device along with the application.
A list of the redistributable shared objects can be found at

```
C:\Program Files (x86)\Intel\System Studio 2015.1.036\
Documentation\en_US\mkl
```

6.1.8 Intel® C++ Compiler dynamic runtime library installation

After unpacking system_studio_target.tgz on the target platform you will find the Intel® C++ Compiler runtime libraries at `../system_studio_target/compiler/lib/<arch>`,

where `<arch>` is `ia32` or `intel64`.

Eclipse* IDE Integration

6.1.9 Installation

The Intel® C++ Compiler for Embedded OS, GDB and SVEN SDK can be automatically integrated into a preexisting Eclipse* CDT installation. The Eclipse* CDK, Eclipse* JRE and the Eclipse* CDT integrated development environment are not shipped with this package of the Intel® System Studio. The Eclipse* integration is automatically offered as one of the last steps of the installation process. If you decide against integration during an earlier install, simply rerun the Intel® System Studio installer.

When asked point the installer to the installation directory of your Eclipse* install. Usually this would be `C:\Program Files (x86)\eclipse\`.

The prerequisites for successful Eclipse integration are:

1. Eclipse* 3.8 (Juno) – Eclipse* 4.3 (Kepler)
2. Eclipse* CDT 8.0 – 8.3
3. Java Runtime Environment (JRE) version 6.0 (also called 1.6) update 11 or later.

Note: The Eclipse* integration of the GDB* GNU Project Debugger requires that the Intel® C++ Compiler installation is selected during Intel® System Studio installation as well.

6.1.10 Launching Eclipse for Development with the Intel C++ Compiler

Since Eclipse requires a JRE to execute, you must ensure that an appropriate JRE is available to Eclipse prior to its invocation. You can set the `PATH` environment variable to the full path of the folder of the `java` file from the JRE installed on your system or reference the full path of the java executable from the JRE installed on your system in the `-vm` parameter of the Eclipse command, e.g.:

```
eclipse -vm \JRE folder\bin\java
```

Invoke the Eclipse executable directly from the directory where it has been installed. For example:

```
<eclipse-install-dir>\eclipse\eclipse
```

6.1.11 Editing Compiler Cross-Build Environment Files

Environment File Support appears under “Intel® System Studio - Intel® System Studio Tools Environment File” on the menu bar.

For details on the Environment File Editor, please check the

Intel® C++ Compiler documentation at

```
<install-dir>\System Studio
2015.1.xxx\Documentation\en_US\compiler_c
```

6.1.12 Cheat Sheets

The Intel® C++ Compiler Eclipse* Integration additionally provides Eclipse* style cheat sheets on how to set up a project for embedded use cases using the Intel® C++ Compiler
In the Eclipse* IDE see

Help > Cheat Sheets > Intel® C/C++ Compiler

6.1.13 SVEN Trace Viewer integration into Eclipse*

Please refer to the SVEN Trace Viewer Installation chapter in

<install-dir>\System Studio 2015.1.xxx\Documentation\en_US\debugger\sven\viewer\SVEN_VIEWER.pdf

6.1.14 Integrating the provided GDB into Eclipse* for remote debug

Remote debugging with GDB using the Eclipse* IDE requires installation of the C\C++ Development Toolkit (CDT) (http://www.eclipse.org/downloads/packages/eclipse-ide-cc-developers/junosr2) as well as Remote System Explorer (RSE) plugins (http:\\download.eclipse.org\tm\downloads\). In addition RSE has to be configured from within Eclipse* to establish connection with the target hardware.

1. Copy the `gdbserver` provided by the product installation
   `(<install-dir>\System Studio 2015.1.xxx\debugger\gdb\<arch>\<python>\bin\)`
   to the target system and add it to the execution PATH environment variable on the target.

2. Configure Eclipse* to point to the correct GDB installation:

a. Inside the Eclipse* IDE click on Window>Preferences from the pulldown menu.

b. Once the preferences dialogue appears select C++>Debug>GDB from the treeview on the left.

c. The GDB executable can be chosen by editing the “GDB debugger” text box. Point to `<install-dir>\System Studio 2015.1.xxx\debugger\gdb\<arch>\<python>\bin\`,

   where `<arch>` is ia32 or intel64 and `<python>` is py24, py26, or py27, depending on architecture and Python* installation

6.1.15 Integrating the Intel® System Debugger into Eclipse*

To add Intel® System Debugger Eclipse* integration after full Intel® System Studio installation or to add the Intel® System Debugger launcher into Wind River* Workbench* this can be done from within Eclipse* by following these steps:

1. Navigate to the `“Help > Install New Software “` entry in the pulldown menu
2. Select `“Add”` and `“Local”` in the following menus …
3. Browse to `<ISS_INSTALL_PATH>\intel_system_debugger_plugins`, where the default for `ISS_INSTALL_PATH` is `C:\Program Files (x86)\Intel\System Studio 2015.1.xxx`

Wind River* Workbench* IDE Integration

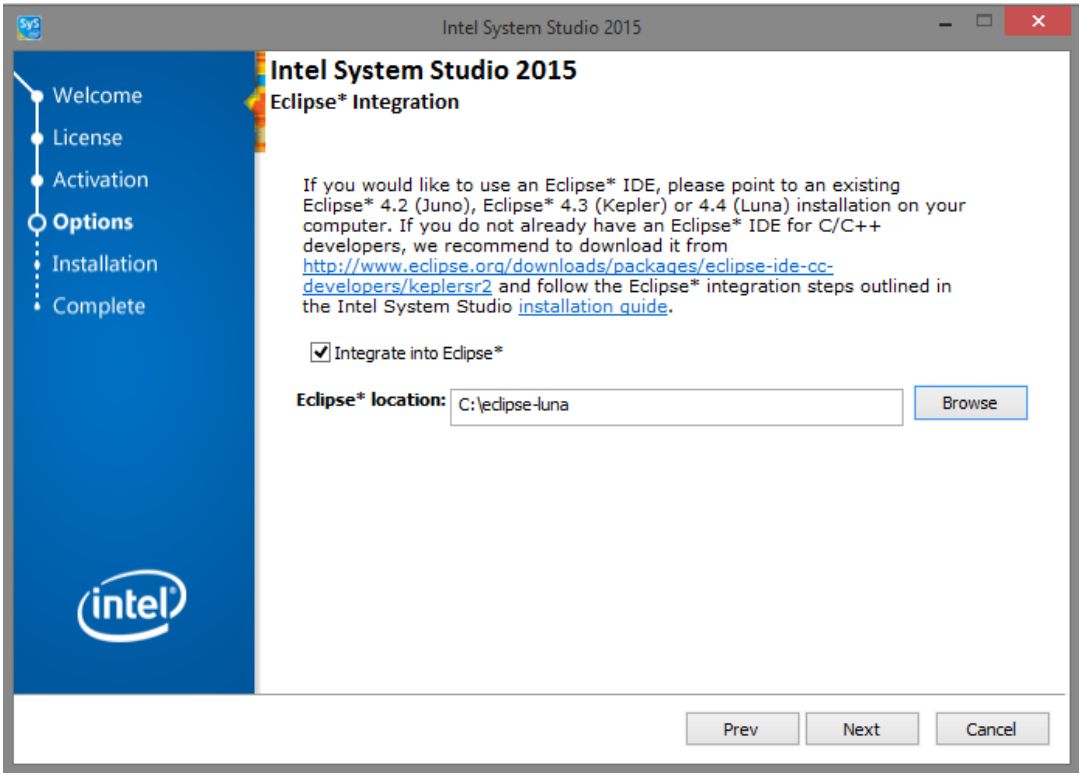
6.1.16 Documentation

1. You will find a detailed `README` file on the integration particulars of Intel® System Studio in the `wr-iss-2015` subdirectory of the Wind River* Workbench* installation directory. This README also goes into the use of the Intel® C++ Compiler as a secondary toolchain layer and adding Intel® System Studio recipes to target platforms for both Wind River* Linux* and Yocto Project*.
2. Additionally there is a Wind River* Workbench integration feature and usage description in the "Using Intel® System Studio with Wind River* Linux* Build Environment" .

6.1.17 Installation

Intel® System Studio provides Wind River* Linux* build environment and platform recipes for Intel® C++ Compiler, Intel® Integrated Performance Primitives, Intel® Math Kernel Library and Intel® VTune™ Amplifier Sampling Collector.
It also integrated IDE launchers for Intel® VTune™ Amplifier for Systems and Intel® System Debugger.

This is offered automatically as a step in the Intel® System Studio product installation:


Intel System Studio 2015
Welcome
License
Activation
Options
Installation
Complete
Intel System Studio 2015
Eclipse* Integration
If you would like to use an Eclipse* IDE, please point to an existing Eclipse* 4.2 (Juno), Eclipse* 4.3 (Kepler) or 4.4 (Luna) installation on your computer. If you do not already have an Eclipse* IDE for C/C++ developers, we recommend to download it from http://www.eclipse.org/downloads/packages/eclipse-ide-cc-developers/keplersr2 and follow the Eclipse* integration steps outlined in the Intel System Studio installation guide.
Integrate into Eclipse*
Eclipse* location:
C:\eclipse-luna
Browse
Prev
Next
Cancel

As part of the installation the following steps are taken implicitly:

1. Create folder `wr-iss-2015` in both the Intel® System Studio installation directory and the Wind River* Workbench* installation directory.
2. In the `wr-setup` subdirectory, execute the script `postinst_wr_iss.bat`. This script will register the platform recipes for different Intel® System Studio components and also the IDE integration of Intel® Systeom components such as Intel® C++ Compiler, Intel® VTune™ Amplifier and Intel® System Debugger.

6.1.18 Manual installation

1. Change into the Wind River* Workbench* installation directory and there into the `..\wr-iss-2015\wr-setup` subdirectory.
2. In the `wr-setup` subdirectory, execute the script `postinst_wr_iss.bat <install-dir>`, providing the Intel® System Studio installation directory as a parameter. This script will register the platform recipes for different Intel® System Studio components and also the IDE integration of Intel® System components such as Intel® C++ Compiler, Intel® VTune™ Amplifier and Intel® System Debugger.

   Note for Windows* host: As the Wind River* Linux* target platform is defined on Linux* host and only imported into the Wind River* Workbench* on Windows* host, Intel® System Studio platform recipes may not be applicable for Windows* host users.

6.1.19 Uninstall

3. Change into the Wind River* Workbench* installation directory and there into the `..\wr-iss-2015\wr-setup` subdirectory.
4. In the `wr-setup` subdirectory, execute the script `uninst_wr_iss.bat`

Installing Intel® XDP3 JTAG Probe

If it is not already pre-installed, the Intel® ITP-XDP3 driver is automatically installed as part of the Intel® System Debugger installation process.

The Intel® ITP-XDP3 driver installer will issue a warning that the publisher could not be verified. Please acknowledge the warning and proceed with the installation.

Installing Macraigor Systems* usb2Demon* Support

The Macraigor Systems* usb2Demon* device can be ordered at http://www.macraigor.com/usbDemon.htm.

To enable support for the Macraigor Systems* usb2Demon* device for debugging Intel® Atom™ processor based platforms with the Intel® System Debugger it is necessary to install the Windows* drivers for the Macraigor Systems* usb2Demon* device. The driver can be found at http://www.macraigor.com/full_gnu.htm.

You can install either the IA-32 or the Intel®64 version of the Hardware Support Package to provide the device driver support

The Intel® System Debugger has been validated for use with the Macraigor Systems* usb2Demon* device and OCDRemote* 12.0-0. We recommend using OCDRemote* 12.0-0 of the Macraigor* Systems* driver for all intended target platforms.

For further details on how to configure the OCDRemote* driver set from Macraigor* Systems, please refer to the full installation instructions at http://www.macraigor.com/full_gnu.htm .

Ordering JTAG Device for Intel® System Debugger

6.1.20 Intel® ITP-XDP3

To order the Intel® ITP-XDP3 device, please

1. either log into your account at https://designintools.intel.com/, select the `Debug Tools` product category and add `ITP-XDP BR3` to your cart.
2. or contact the Hibbert Group* at Intelvtg@hibbertgroup.com and request the VTG order form.

We will also gladly assist with the ordering process. If you have any questions please submit an issue in the Intel® System Studio product of Intel® Premier Support https://premier.intel.com or send an email to IntelSystemStudio@intel.com.

6.1.21 Macraigor* usb2Demon*

Go to http://www.macraigor.com/usbDemon.htm and select the Intel(R) Atom™ Processor target with the appropriate 24, 31 or 60 pin connector for your target device.

Removing the Product

To uninstall the Intel® System Studio use the Windows* Control Panel Program uninstall feature.

7 Issues and Limitations

Known Issues and Limitations

For known issues of individual Intel® System Studio components please refer to the individual component release notes. Their location in the installed product can be found in chapter 2:

Technical Support and Documentation

7.1.1 Documentation Links and FAT32 file system

If Intel® System Studio is installed on a FAT32 file-system, the symbolic links for Intel® VTune™ Amplifier for Systems and Intel® Inspector for Systems documentation from `C:\Program Files (x86)\Intel\System Studio 2015.y.xxx\`Documentation\ may not work.

Please refer directly to the

```
C:\Program Files (x86)\Intel\System Studio 2015.y.xxx\VTune Amplifier 2015 for Systems\Documentation\
```

and

```
C:\Program Files (x86)\Intel\System Studio 2015.y.xxx\Inspector 2015 for Systems\Documentation\
```

7.1.2 Running online-installer behind proxy server fails

Running online-installer behind proxy server produces error: "Connection to the IRC site cannot be established". Please see the Installation Notes for more details

Graphics Analysis Tools

7.1.3 Installation failure on Windows* host with script custom actions

The installation of Intel® System Studio System Analyzer, Frame Analyzer and Platform Analyzer may fail on rare occasions with the following Windows* error message:

1. 2738, Could not access VBScript run time for custom action [2].
2. 2739, Could not access JScript run time for custom action [2].

If this error message occurs, the installation can be completed after applying the following steps:

- Check that vbscript.dll and jscript.dll aren't registered in HKEY_CURRENT_USER (HKCU), checking for the registry keys below.

- VBScript, HKCU\SOFTWARE\Classes\CLSID\{ B54F3741-5B07-11CF-A4B0-00AA004A55E8}
- JScript, HKCU\SOFTWARE\Classes\CLSID\{ F414C260-6AC0-11CF-B6D1-00AA00BBBB58}
- JScript, HKCU\SOFTWARE\Classes\CLSID\{ F414C261-6AC0-11CF-B6D1-00AA00BBBB58}
- JScript, HKCU\SOFTWARE\Classes\CLSID\{ F414C262-6AC0-11CF-B6D1-00AA00BBBB58}
- Remove these keys if they exist in HKEY_CURRENT_USER..

7.1.4 Support for Intel® Atom™ Processor Z3560 and Z3580 code-named "Moorefield" missing

Support for Intel® Atom™ Processor Z3560 and Z3580 code-named "Moorefield" is currently not available

Intel® Energy Profiler

7.1.5 \boot\config-'uname –r' file must be present on platform.

In order to enable CPU power data collection for Intel® VTune™ Amplifier please make sure your environment does have a file named `\boot\config-`uname -r`` located in your `\boot\config` directory

If there is no such file you should run the following command:

```
$ cat \proc\config.gz | gunzip - > \boot\config-`uname -r`
```

7.1.6 Power and Frequency Analysis support for Intel® Atom™ Processor covers Android* OS only.

Power and frequency analysis currently requires at least a 2nd generation Intel® Core™ Processor Family based platform or an Intel® Atom™ Processor Z2xxx or Z3xxx running Android* OS

Intel® VTune™ Amplifier Usage with Yocto Project*

7.1.7 Building Sampling Collector (SEP) for Intel® VTune™ Amplifier driver on host Linux* system

For Yocto Project* targeted development additional kernel utilities required for building drivers and kernel modules need to be present in the kernel source tree. The following utilities need to be manually added to the standard Yocto Project* 1.x kernel build tree: viz, recordmcount, fixdep, and modpost.

7.1.8 Remote Intel® VTune™ Amplifier Sampling on Intel® 64 Yocto Project* Builds

The GNU linker ld is installed in a non-standard path on Yocto Project* 1.5 for Intel® 64 (x86_64). For remote sampling with amplxe-runss to work correctly "/lib64/ld-linux-x86-64.so.2 " has to be added as a symlink to /lib/ld-linux-x86-64.so.2 on the target filesystem.

7.1.9 Building 64bit Sampling Collector against Yocto Project* targeting Intel® Atom™ Processor E38xx requires additional build flags

Building the Intel® VTune™ Amplifier for Systems Sampling Collector driver SEPDK against the x86_64 version of Yocto Project 1.6 (Daisy) for Intel® Atom™ Processor E38xx requires a modification of the `Makefile` in `../sepdk/src` and `../sepdk/pax`.

In both cases the `EXTRA_CFLAGS` entry needs to be amended with the option `-DCONFIG_COMPAT`:

```
EXTRA_CFLAGS += -I$(LDDINCDIR) -I$(LDDINCDIR1) -DCONFIG_COMPAT
```

Intel® System Debugger

7.1.10 Intel® Puma™ 6 Media Gateway Firmware Recovery Tool not available

The start_xdb_firmware_recovery.sh / start_xdb_firmware_recovery.bat utility to allow recovery of corrupted flash memory on the Intel® Puma 6 Media Gateway is not functional in the Intel® System Debugger 2015 Beta.

7.1.11 Using the symbol browser on large data sets and large symbol info files not recommended

It is recommended to use the source files window to browse to the function to debug instead of the symbol browser as the use of the symbol browser on large data sets and large symbol information files (e.g. Android* kernel image) can lead to debugger stall.

7.1.12 Limited support for Dwarf Version 4 symbol information

If when debugging binaries generated with GNU* GCC 4.8 or newer the line information and variable resolution in the debugger is unsatisfactory, please try to rebuild your project using the `-gdwarf-3` option instead of simply `-g`.

GDB* - GNU* Project Debugger

7.1.13 Eclipse* integration of GDB* requires Intel® C++ Compiler install

The Eclipse* integration of the GDB* GNU Project Debugger requires that the Intel® C++ Compiler installation is selected during Intel® System Studio installation as well.

Software Visible Event Nexus (SVEN)

7.1.14 SVEN SDK is not targeting Intel® Atom™ Processor CE series

The SVEN SDK included in this distribution is not intended for use with the Intel® Atom™ Processor CE41xx, CE42xx, CE53xx, or the Intel® Puma™ 6 Media Gateway.

To take advantage of all the capabilities provided by SVEN on those platforms, please contact your Intel Field representative.

Intel® Integrated Performance Primitives

7.1.15 Some Intel® IPP domains are not installed by default

Several Intel® IPP domains (ippRR, ippGEN, ippJP, ippAC, ippVC, and ippSC) are not installed by default. Please use the Intel® System Studio installation customization option to add them.

Intel® C++ Compiler

7.1.16 "libgcc_s.so.1" should be installed on the target system

By default the Intel® C++ Compiler links the compiled binary with the library "libgcc_s.so.1". Some embedded device OSs, for example Yocto-1.7, don't have it in default

8 Attributions

This product includes software developed at:

The Apache Software Foundation (http:\\www.apache.org\).

Portions of this software were originally based on the following:
- software copyright (c) 1999, IBM Corporation., http:\\www.ibm.com.
- software copyright (c) 1999, Sun Microsystems., http:\\www.sun.com.
- the W3C consortium (http:\\www.w3c.org) ,
- the SAX project (http:\\www.saxproject.org)
- voluntary contributions made by Paul Eng on behalf of the
Apache Software Foundation that were originally developed at iClick, Inc.,
software copyright (c) 1999.

This product includes updcrc macro,
Satchell Evaluations and Chuck Forsberg.
Copyright (C) 1986 Stephen Satchell.

This product includes software developed by the MX4J project
(http:\\mx4j.sourceforge.net).

This product includes ICU 1.8.1 and later.
Copyright (c) 1995-2006 International Business Machines Corporation and others.

Portions copyright (c) 1997-2007 Cypress Semiconductor Corporation.
All rights reserved.

This product includes XORP.
Copyright (c) 2001-2004 International Computer Science Institute

This product includes software licensed from Macraigor Systems, LLC.
Copyright (c) 2004-2009, Macraigor Systems LLC. All rights reserved.

This product includes software from the book
"Linux Device Drivers" by Alessandro Rubini and Jonathan Corbet,
published by O'Reilly & Associates.

This product includes hashtab.c.
Bob Jenkins, 1996.

9 Disclaimer and Legal Information

The Intel® C++ Compiler, Intel® Integrated Performance Primitives, Intel® Math Kernel Library, Intel® VTune™ Amplifier, and Intel® Inspector for Systems are provided under Intel's End User License Agreement (EULA).

The GNU* Project Debugger, GDB is provided under the General GNU General Public License GPL V3.

The SVEN SDK is provided under the General GNU General Public License GPL V2.

The SVEN Trace Viewer is provided under the Eclipse Public License EPL V1.

The Intel® System Debugger kernel module xdbntf.ko is provided under the General GNU General Public License GPL V2.

Please consult the licenses included in the distribution for details.

INFORMATION IN THIS DOCUMENT IS PROVIDED IN CONNECTION WITH INTEL PRODUCTS. NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. EXCEPT AS PROVIDED IN INTEL'S TERMS AND CONDITIONS OF SALE FOR SUCH PRODUCTS, INTEL ASSUMES NO LIABILITY WHATSOEVER, AND INTEL DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO SALE AND\OR USE OF INTEL PRODUCTS INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT.
UNLESS OTHERWISE AGREED IN WRITING BY INTEL, THE INTEL PRODUCTS ARE NOT DESIGNED NOR INTENDED FOR ANY APPLICATION IN WHICH THE FAILURE OF THE INTEL PRODUCT COULD CREATE A SITUATION WHERE PERSONAL INJURY OR DEATH MAY OCCUR.
Intel may make changes to specifications and product descriptions at any time, without notice. Designers must not rely on the absence or characteristics of any features or instructions marked "reserved" or "undefined." Intel reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them. The information here is subject to change without notice. Do not finalize a design with this information.
The products described in this document may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request.
Contact your local Intel sales office or your distributor to obtain the latest specifications and before placing your product order.
Copies of documents which have an order number and are referenced in this document, or other Intel literature, may be obtained by calling 1-800-548-4725, or go to: http:\\www.intel.com\design\literature.htm

Intel processor numbers are not a measure of performance. Processor numbers differentiate features within each processor family, not across different processor families. Go to: http:\\www.intel.com\products\processor_number\

MPEG-1, MPEG-2, MPEG-4, H.261, H.263, H.264, MP3, DV, VC-1, MJPEG, AC3, AAC, G.711, G.722, G.722.1, G.722.2, AMRWB, Extended AMRWB (AMRWB+), G.167, G.168, G.169, G.723.1, G.726, G.728, G.729, G.729.1, GSM AMR, GSM FR are international standards promoted by ISO, IEC, ITU, ETSI, 3GPP and other organizations. Implementations of these standards, or the standard enabled platforms may require licenses from various entities, including Intel Corporation.

BunnyPeople, Puma, Celeron, Celeron Inside, Centrino, Centrino Inside, Cilk, Core Inside, i960, Intel, the Intel logo, Intel AppUp, Intel Atom, Intel Atom Inside, Intel Core, Intel Inside, Intel Inside logo, Intel NetBurst, Intel NetMerge, Intel NetStructure, Intel SingleDriver, Intel SpeedStep, Intel Sponsors of Tomorrow., the Intel Sponsors of Tomorrow. logo, Intel StrataFlash, Intel Viiv, Intel vPro, Intel XScale, InTru, the InTru logo, InTru soundmark, Itanium, Itanium Inside, MCS, MMX, Moblin, Pentium, Pentium Inside, skoool, the skoool logo, Sound Mark, The Journey Inside, vPro Inside, VTune, Xeon, and Xeon Inside are trademarks of Intel Corporation in the U.S. and other countries.

* Other names and brands may be claimed as the property of others.

Microsoft, Windows, Visual Studio, Visual C++, and the Windows logo are trademarks, or registered trademarks of Microsoft Corporation in the United States and\or other countries.

Java is a registered trademark of Oracle and\or its affiliates.

Copyright (C) 2008–2014, Intel Corporation. All rights reserved.